DPfeature: feature selection by detecting DEGs across cluster of cells

Monocle 2 selects genes to be used for constructing a trajectory and ordering cells along it. First, dpFeature filters genes that only expressed in a very small percentage of cells (by default, 5%). Second, dpFeature performs PCA on the expressed genes and allows the user to decide the number of principal components (PCs) used for downstream analysis, generally by identifying the first PC that fails to explain a substantial amount of marginal variance. The top PCs will be further used to initialize t-SNE which projects the cells into two-dimensional t-SNE space. Third, dpFeature clusters the cells in the t-SNE space using density peak clustering [1]. The density peak clustering algorithm calculates each cell's local density (ρ) and the distance (δ) between it and its nearest neighboring cell with higher local density. Cells with high local density that are far away from other cells with higher local density correspond to the density peaks. These density peaks nucleate clusters: all other cells will be associated with the nearest density peak cell to form clusters. Finally, dpFeature identify genes that differ between the clusters by performing a likelihood ratio test between a negative-binomial generalized linear model that knows the cluster to which each cell is assigned and a model that doesn't. By default, up to 1,000 of the most significant genes ($FDR < 10\%$) after Benjamini-Hochberg correction are reported as the ordering genes.

Simulating the neuronal differentiation trajectory

Previously, Qiu et al used a system of differential equations describing the dynamics of 12 genes [2] to simulate the hypothetical differentiation process of three cell types from the central nervous system. We use this approach to generate idealized, branched differentiation trajectories for benchmarking Monocle 2 (The nuisance factor Mature from the orignal study is set to zero as this was found to help the system avoid ending up in the chaotic regime). The network mainly consists of two mutual inhibition TF-pairs where the first one (between $Mash1$ and $Hes5$) specifies the bifurcation between neuron and glia, the second one (between Scl and $Olig2$) specifies the bifurcation between astrocyte and oligodendrocyte. All genes are initialized to zero except Pax3. We then follow the gene expression dynamics of all 12 genes by numerically calculating the output of the stochastic differential equation (SDE) for each gene over 400 time steps in a time span of 20 using the Euler method [3]. Random noise injected into the simulation at each time step leads to the spontaneous, nondeterministic selection of one of three cell fates each time the simulation is run. We therefore run the simulation 200 times to ensure that we obtain a population of cells with representative paths leading from the root to each fate. The gene expression of master regulators ($Mash1$, $Hes5$, Scl and $Olig2$) at the last time step for each simulated developmental trajectory are used to classify trajectories into either neuron, astrocyte or oligodendrocyte type. Similarly, branch assignments are determined by manual inspection of bifurcation point of master regulator pairs (Mash1-Hes5, Scl-Olig2) in the simulation.

We can also obtain a theoretical trajectory from the SDEs as follows. First, we calculate the stable fixed points and metastable ones (saddle points) for the system, which correspond to the progenitor or terminal cell states and the bifurcation points of the differentiation process. According to the A-type integration view of stochastic dynamics, fixed points of their ODE counterparts can be directly regarded as most probable states of SDE [4, 5]. That is, the least action paths [6, 7] (in the sense of dynamical systems) connecting the fixed point corresponding to the progenitor cell state to the other fixed points, corresponding to neuron, glia and oligodendrocyte fate, of the analogous system of ODEs define the most probable path through the SDEs [7]. We take these

paths as the theoretical trajectory.

To assess Monocle 2's performance of principal graph learning on the simulation data, we selected a simulated trajectory for each of the cell fates, and provided Monocle 2 with the expression values for each of the 12 genes sampled at all time steps. We then run Monocle 2 with default parameters. We also ran Monocle 2 using SimplePPT instead of DDRTree. DDRTree learns the trajectory in the a two-dimensional latent space while SimplePPT learns in the original 12 dimension. The trajectories produced by Monocle 2 were then plotted in the high-dimensional space (SimplePPT, **SupplImentary Figure 3H-I**) or the reversed-embedding dimension one (**Supplementary Figure 3F-G**) by projecting low dimension data back to the original dimension (DDRTree) against the theoretical trajectory.

In DDRTree, reversed embedding is achieving by multiplying **W** matrix to the coordinates of all reduced dimensional points (matrix $\mathcal{Z}$) , that is, $\mathbf{W} \cdot \mathcal{Z}$. This method may be useful to recover the denoised high dimensional data (see function $reverseEnbeddingCDS$).

Initializing RGE with different dimension reduction methods

We implement PCA, ICA, LLE, diffusion maps and pass each function to reduceDimension in Monocle 2 to assess the robustness of different RGE techniques (DDRTree, SimplePPT, SGL) in pseudotime estimation and branch assignment under different dimension reduction initialization approach. PCA is performed with prcomp function from R [8]. ICA is based on ica_helper function from Monocle 2 package where ica_helper function is built on irlba package [9]. For LLE, we identified the best number of neighbors based on the $select\_k$ function from SLICER and then run lle function from lle package with the identified k while keeping all other parameters as default [10]. For diffusion maps, we run diffusionMap from diffusionMap package [11] with default parameters on the distance matrix of the input normalized expression data. We use HSMM dataset for this benchmark analysis. SimplePPT, DDRTree, SGL are run with default parameters, excepting gamma is set as 0.005 for SGL-tree and 0.1 for $\mathcal{L}_1$ graph.

Practical suggestions on reconstructing single-cell RNA-seq dataset with Monocle 2

Users can often simply run Monocle 2 by default setting which initializes dimension reduction with PCA and applies DDRTree as for reverse graph embedding. The principal graph will be constructed in first two dimensions while the number of principal points ($ncenter$) will be automatically chosen. We used this setting in HSMM, lung, simulation as well as MARS-seq benchmark analysis. We recommend users to check the scree plot (see $plot\_pc\_variance\_explained$ function) to decide whether or not a higher number of dimensions is desirable for RGE (based on a significant drop in variance explained by the PC component). A higher number of dimension can be used to detect more intricate branches. We used higher dimension for trajectory reconstruction in Olsson ($max\_component = 4$) and Paul ($max\_component = 10$) datasets analysis. Although the tree is constructed in high dimension, it can be visualized in two dimension using a tree layout. This layout preserves the graph structure, branch assignments, and the relative ordering of cells but does not preserve their pairwise euclidean distance in the reduced-dimensional space.

As more advanced dimension reduction methods emerge, users are welcome to apply RGE (SimplePPT, SGL-tree, L1-graph) directly on the reduced dimension space for principal graph learning and trajectory reconstruction.

The automatic determination of ncenter parameter applied in Monocle 2 often regularizes the data and clears all insignificant branches. However, there are cases where tiny branches still exists which implies those small fractions of cells deviates from the structure of major cell progression. Users may ignore those tiny branches (either for biological discovery of major transcriptomic changes or if they are interested in benchmarking Monocle 2 with software requiring manual specification of branch numbers), a convenient trimming procedure can be applied to merge those branches into nearby tree segment (see $trimTree$ function) based on the learned principal graph.

Feature selection prior trajectory reconstruction ensures Monocle 2 to reconstruct the developmental trajectory more accurately, users are recommended to first apply dpFeature procedure to obtain the ordering genes and then reconstruct developmental trajectory.

Scalability and complexity analysis of Monocle 2

Monocle 2 is scalable to analyze large scale droplet based single cell RNA-seq datasets (or datasets based on other new technologies, including, sci-RNA-seq [12]). The complexity of different implementation of RGE is discussed in the original papers ([13–15]) and summarized in **Supplementary Figure 13B**. Briefly, DDRTree's complexity ($O(K^3 + D^3 + DK^2 + ND^2 + NDK)$) can be approximated as $O(ND^2)$ where N is the number of cells, D is the number of feature genes and K is the number of centroid which is normally smaller than 200 defined by our automatic centroid number selection procedure. This complexity is on the same scale as DPT or Wishbone (roughly both of them have complexity $O(Nk^2)$ where N is the number of cells and K is the number of nearest neighbors, see [16]). We compared the running time of Monocle 2, DPT and Wishbone on the full Paul dataset (8365 cells in total). Monocle 2 performs similarly to DPT but Wishbone is more efficient in terms of running time with the caveat that Monocle 2 maintains various auxiliary datasets used for downstream analysis, for example BEAM, etc.

Trajectory-conditioned test

The trajectory-conditioned test is a new type of DEG (differential gene expression) test designed to identify genes that differ between cells from different categories (for example, a mutant and the wild type), controlling for where the cells are on the trajectory. For test in **Supplementary Figure 19**, we first identify the range of the pseudotime of the knockout cells on a particular branch. Wild-type cells among this pseudotime range on the same branch are then selected. The knockout cells or the wild-type cells are pooled as two different groups and then a two-group test (because we only have small number of cells for each phenotype category) is performed.

In principle, trajectory-conditioned test can run by considering both of the pseudotime and the grouping, similarly to the previously defined BEAM test, where the full model is

$$g(E(Y)) = \beta_0 + f_1(G) + f_2(\varphi)$$

The alternative model is

$$g(E(Y)) = \beta_0 + f_1(G)$$

In each of the model, $E(Y)$ represents the expected value for the transcript counts data Y where Y is negative binomial distributed or $Y \sim NB(r, p)$. g is a link function; for negative binomial distribution, it is log. $f_1(G)$ represents the indicator function for the genotype G while $f_2(\varphi)$ represents the non-parameteric function, such as the natural spline implemented in $VGAM$ [17] ($sm.ns$ function), of pseudotime φ.

In practice, this general strategy requires there be enough cells in each category to accurately fit the pseudotime spline curves but will likely be more powered.

Details on analyzing datasets used in this study

Analysis of HSMM data

Human skeletal muscle myoblast (HSMM) data from our previous publication [18] is used and converted into transcript counts using Census [19] with default parameters. We first filter cells with total Census counts larger than $1e^6$, then filter cells with total Census counts above or below two standard deviation of total Census counts. Potential contaminating fibroblast cells are then identified and removed using a semi-supervised approach, as descibed in the Monocle vignette. We then run dpFeature to select ordering genes (the top $1,000$ DEGs are used) on the muscle cells. The top four PC components are used, ρ and δ are set to 2.6 and 4, respectively based on the decision plot to cluster cells into four clusters using the density peak clustering algorithm [1].

As previously described [19], the ward.D2 clustering method is then applied on the correlation matrix for the row-scaled data (also truncated at 3 or -3) for the Census counts for the $1,000$ feature genes between all the cells. To obtain the enriched GO BP terms for the **Supplementary Figure 1D**, we perform a hypergeometric test on the corresponding Gene Matrix Transposed file format (GMT) file for each cluster of genes based on the piano package [20]. For PCA based feature selection, we select top 1,000 genes with highest loading from the first two principal components. For SLICER's feature selection, SLICER is run with default parameters. For high dispersion based feature selection, genes with mean expression larger than 0.1 and empirical dispersion larger than the theoretical dispersion based on empirical mean expression. We used DESeq's method [21] to fit a relationship between mean expression and the variance using a gamma function. For feature selection with DEG test based on known cell group labels, we used the information of collected time point (0 hour, 24 hour, 48 hour or 72 hour) for each cell and identify DEGs between cells from different time points based on generalized linear models and likelihood-ratio test, implemented in Monocle.

To reconstruct the myogenesis trajectory, Monocle 2 is used with default parameters.

Analysis of lung data

Lung data is processed as described previously [19]. We used transcript counts estimated with spike-in for all analysis of this dataset. We then run dpFeature to select ordering genes (top $1,000$ DEGs are used) of the 183 cells. Top five PC components are used, ρ and δ is set to 3 and 9 based on the decision plot to cluster cells into four clusters. PCA, SLICER, high dispersion based feature selection are done as described above. For DEG based feature selection, time labels of $E14.5$ days, $E16.5$ days, $E18.5$ days and Adult AT2 cells are used.

Analysis of Paul data

UMI counts data for the Paul experiment [22] as well as annotation of cells, etc. are downloaded from http://compgenomics.weizmann.ac.il/tanay/?page_id=649. Cells assigned to clusters in the original study are used to classify cells. For benchmarking accuracy and robustness, we combined clusters $1, 2, 3, 4, 5, 6$ as erythroid (1095), clusters $7, 8, 9, 10$ as CMP (451), clusters $11, 12, 13, 14, 15, 16, 17, 18$ as GMP (1123) and cluster 19 as lymphoid cells (31), as suggested in

the original study [22]. The CMP and GMP cells can be further classified into, megakaryocyte and erythrocyte progenitor (MEP, cluster 7), Megakaryocyte (MK, cluster 8), early monocyte / granulocyte progenitor (GMP, cluster 9, 10), dendritic cell (DC, cluster 11, 30 cell), basophil (Bas, cluster 12, 13), monocyte (Mon, cluster 14, 15), neutrophil (Neu, cluster 16, 17). To ensure better comparison with other published work [16, 23], we used all valid informative genes (3004 genes) identified in the original study [22] for cell ordering. But we also run dpFeature to select ordering genes (top $1,000$ DEGs are used) on the transcript counts in **Supplementary Figure SI2H**. Top five PC components are used, ρ and δ are set to 3 and 40 based on the decision plot to cluster cells into three clusters. PCA, SLICER, high dispersion based feature selection are done as described above. For DEG based feature selection, cell type labels (CMP, GMP and lymphoid cells) annotated as above are used. We run Monocle 2 for trajectory reconstruction with default parameters.

Lymphoid cells are removed from our analysis because they belong to a different developmental lineage.

Analysis of the downsampling dataset with DPT, Wishbone, SLICER

We run DPT (destiny 2.0.3), Wishbone (https://github.com/ManuSetty/wishbone), SLICER (SLICER 0.2.0) using all the same informative genes from the original study [22] as ordering genes. For Monocle 2, we set $norm\_method =' log'$ which will log transform the expression value (after adding a pseudocount 1, same as below) and then run orderCells function with default parameters. For Monocle 1, we set $norm\_method =' log'$, $reduction\_method =' ICA'$ in reduceDimension function and set $num\_paths = 2$ in the orderCells function. For DPT, we also first log transform the expression value and then run DPT with default parameters. Since Wishbone requires manually selecting the diffusion map components used for trajectory construction which prevent automatic benchmarking, the first two reduced diffusion components from DPT are thus passed into Wishbone (we also tried using the first five diffusion dimension calculated in Wishbone which generally gives worse results). For SLICER, we again first log transform the expression value, then use the $select\_k$ function to select the best number of neighbors used in lle function. min_branch_len was set to 10 when running $assign\_branches$ function. Root cells were properly selected for each software to ensure pseudotime calculation for all software starts from the same cells. Both Monocle 2, Monocle 1 and SLICER reduce the high-dimension data into two intrinsic dimensions.

Analysis of Olsson data

FPKM values for the blood study is downloaded from synapse (id $syn4975060$). Cell types or gates information are downloaded from the online data along with the original study. We then run Census to estimate the transcript counts. For **Supplementary Figure 17**, wild-type cells (excepting the two transition gate cells) are used. We run dpFeature to select ordering genes (top $1,000$ DEGs are used) on the Census counts. Top five PC components are used, ρ and δ are set to 6.4 and 8.1 based on the decision plot to cluster cells into three clusters. PCA, SLICER, high dispersion based feature selection are done as described above. For DEG based feature selection, cell type labels based on the original study are used.

We run BEAM to obtain genes significantly branching between granulocyte or monocyte lineages in the wild type blood dataset [24]. Similarly to previously described [19], those BEAM genes are then used to create the branch heatmap in **Supplementary Figure 17C** where gene expression pattern along the granulocyte or monocyte branch is clustered into 8 clusters. To obtain the enriched GO BP terms for the **Supplementary Figure 17C**, we perform the hypergeometric test on the corresponding Gene Matrix Transposed file format (GMT) file for each cluster of genes

based on the piano package [20]. ChIP-seq data for Gfi1 and Irf8 are downloaded and analyzed using MACS2 [25] to identify the potential targets. DHS dataset ($GSE59992$) for the GMP cells are downloaded [26]. We then use FIMO [27] to scan the DHS peaks for the JASPAR vertebrate motif database (version 2016) [28], which give us the regulatory relationship between TFs (with the motif in JASPAR database) and their potential targets. We only focus on those TFs which are both significant BEAM genes and are potentially bound by either Gfi1 or Irf8 based on the ChIP-seq data. We define those TFs as the potential direct targets of master regulator Gfi1 and Irf8. Then we look for the significant BEAM genes which are potentially targeted by those direct targets based on FIMO scanning and define them as the secondary targets. By applying methods discussed previously [19], we calculate the branch time point for all BEAM genes and categorize those genes either as master regulators, direct or secondary targets or other BEAM genes which are not included in all previous sets.

For figure 6, we used the same $1,000$ genes selected just from wild-type cells to order the entire dataset, including the wild-type data, the transition gate cells as well as the $Irf8^{-/-}$, $Gfi1^{-/-}$ or $Irf8^{-/-}Gfi1^{-/-}$ cells.

Benchmarking Monocle 2 on the simulation dataset

For benchmarking the accuracy of Monocle 2 on the simulation dataset (400 time points for either neuron or astrocyte lineage are used), we used the time index from the simulation as real time. Two branch points are determined based on inspection of the bifurcation point of the kinetic curves of Mash1-Hes5 pair and Scl-Olig2 pair over the simulation. We found DPT estimates pseudotime excellently which may relate to its theoretical relationship to stochastic differential equations we used for simulating this dataset [29]. However it tends to assign cells near real bifurcation point as metastable cells less robustly (same cell assigned as metastable or other state under different downsampling cases randomly) which implies its incapability in structural learning without an explicit underlying assumption of graph structure (We avoid considering those metastable cells when perform benchmarking). Although Monocle 2 has higher robustness in pseudotime and branch assignment, its accuracy is slightly worse than DPT. Wishbone performs moderately on this simulation dataset.

References

1. Rodriguez, A. & Laio, A. Clustering by fast search and find of density peaks. en. *Science* **344,** 1492–1496 (27 06 2014).
2. Qiu, X., Ding, S. & Shi, T. From understanding the development landscape of the canonical fate-switch pair to constructing a dynamic landscape for two-step neural differentiation. en. *PLoS One* **7,** e49271 (20 12 2012).
3. Atkinson, K. E. *An introduction to numerical analysis* (John Wiley & Sons, 2008).
4. Shi, J., Chen, T., Yuan, R., Yuan, B. & Ao, P. Relation of a New Interpretation of Stochastic Differential Equations to Ito Process. en. *J. Stat. Phys.* **148,** 579–590 (Jan. 2012).
5. Ascher, U. M. & Petzold, L. R. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations* en (SIAM, Jan. 1998).
6. Freidlin, M. I. & Wentzell, A. D. *Random Perturbations of Dynamical Systems:* (Springer New York, 1998).

7. Tang, Y., Yuan, R., Wang, G., Zhu, X. & Ao, P. Potential landscape of high dimensional nonlinear stochastic dynamics and rare transitions with large noise. arXiv: 1611.07140 [cond-mat.stat-mech] (22 11 2016).
8. Team, R. C. *R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. 2013* 2014.
9. Baglama, J. & Reichel, L. Augmented Implicitly Restarted Lanczos Bidiagonalization Methods. *SIAM J. Sci. Comput.* **27,** 19–42 (2005).
10. Wang, H., Zheng, J., Yao, Z. & Li, L. in *Advances in Neural Networks - ISNN 2006* (eds Wang, J., Yi, Z., Zurada, J. M., Lu, B.-L. & Yin, H.) 1326–1333 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2006).
11. Richards, J. W., Freeman, P. E., Lee, A. B. & Schafer, C. M. EXPLOITING LOW-DIMENSIONAL STRUCTURE IN ASTRONOMICAL SPECTRA. en. *ApJ* **691,** 32 (July 2009).
12. Cao, J. *et al. Comprehensive single cell transcriptional profiling of a multicellular organism by combinatorial indexing* en. Feb. 2017.
13. Mao, Q., Yang, L., Wang, L., Goodison, S. & Sun, Y. in *Proceedings of the 2015 SIAM International Conference on Data Mining* 792–800 ().
14. Mao, Q., Wang, L., Goodison, S. & Sun, Y. *Dimensionality Reduction Via Graph Structure Learning* in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (ACM, New York, NY, USA, 2015), 765–774.
15. Mao, Q., Wang, L., Tsang, I. & Sun, Y. Principal Graph and Structure Learning Based on Reversed Graph Embedding. en. *IEEE Trans. Pattern Anal. Mach. Intell.* (May 2016).
16. Haghverdi, L., Buettner, M., Alexander Wolf, F., Buettner, F. & Theis, F. J. *Diffusion pseudotime robustly reconstructs lineage branching* en. Jan. 2016.
17. Yee, T. The VGAM Package for Categorical Data Analysis. *J. Stat. Softw.* **32,** 1–34 (2010).
18. Trapnell, C. *et al.* The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. en. *Nat. Biotechnol.* **32,** 381–386 (Apr. 2014).
19. Qiu, X. *et al.* Single-cell mRNA quantification and differential analysis with Census. en. *Nat. Methods* (23 01 2017).
20. Väremo, L., Nielsen, J. & Nookaew, I. Enriching the gene set analysis of genome-wide data by incorporating directionality of gene expression and combining statistical hypotheses and methods. en. *Nucleic Acids Res.* **41,** 4378–4391 (Apr. 2013).
21. Anders, S. & Huber, W. Differential expression analysis for sequence count data. en. *Genome Biol.* **11,** R106 (27 10 2010).
22. Paul, F. *et al.* Transcriptional Heterogeneity and Lineage Commitment in Myeloid Progenitors. en. *Cell* **163,** 1663–1677 (17 12 2015).
23. Setty, M. *et al.* Wishbone identifies bifurcating developmental trajectories from single-cell data. en. *Nat. Biotechnol.* **34,** 637–645 (June 2016).
24. Olsson, A. *et al.* Single-cell analysis of mixed-lineage states leading to a binary cell fate choice. en. *Nature* **537,** 698–702 (29 09 2016).
25. Zhang, Y. *et al.* Model-based analysis of ChIP-Seq (MACS). en. *Genome Biol.* **9,** R137 (17 09 2008).

26. Lara-Astiaso, D. *et al.* Immunogenetics. Chromatin state dynamics during blood formation. en. *Science* **345,** 943–949 (22 08 2014).

27. Grant, C. E., Bailey, T. L. & Noble, W. S. FIMO: scanning for occurrences of a given motif. en. *Bioinformatics* **27,** 1017–1018 (Jan. 2011).

28. Mathelier, A. *et al.* JASPAR 2016: a major expansion and update of the open-access database of transcription factor binding profiles. en. *Nucleic Acids Res.* **44,** D110–5 (Apr. 2016).

29. Nadler, B., Lafon, S., Kevrekidis, I. & Coifman, R. R. *Diffusion maps, spectral clustering and eigenfunctions of Fokker-Planck operators* in *Advances in neural information processing systems* (2006), 955–962.